Synthesis of Series-Parallel Network Switching Functions

By WARREN SEMON*

(Manuscript received February 13, 1958)

From the switching functions of n *variables, those which correspond to networks are abstracted and called network functions. Properties of those network functions corresponding to series-parallel networks are studied and a method for synthesis is developed.*

I. INTRODUCTION

It is known that one may establish a mathematical model in which Boolean polynomials explicitly represent two terminal single-impulse series-parallel contact networks. The conventions used in this paper are Boolean plus, symbolized by $\oplus$, to represent parallel connection; Boolean times, symbolized by $\cdot$ or by juxtaposition, to represent series connection; zero, to represent an open circuit; and one, to represent a closed circuit. The symbol $'$ is used to represent Boolean negation, also called "inversion" or "complementation." Whenever f represents an open circuit, f' represents a closed circuit and conversely. This is a conductance analogue, the dual of that used by Shannon.[1] If $f(x_1, x_2, \cdots, x_n)$ is a switching function it may be represented in tabular form[2] as a "canonical-form matrix" whose rows consist of those configurations of the variables for which $f = 1$, as shown below for the function

$$f = x_1'(x_2' \oplus x_3') = x_1'x_2'x_3' \oplus x_1'x_2'x_3 \oplus x_1'x_2x_3'.$$

x_1	x_2	x_3
0	0	0
0	0	1
0	1	0

* The Computation Laboratory of Harvard University, Cambridge, Mass.

Definition 1

A "series-parallel" network is a series or parallel combination of two series-parallel networks; a single element is a series-parallel network.[3]

Definition 2

A "network" function is a switching function of n *variables, having no vacuous variables, which can be realized by a network containing* n *switches.*

As an example: $f(x_1, x_2, x_3, x_4) = x_1x_2' \oplus x_3'x_4$ is a network function. A four-switch realization is shown in Fig. 1.

Definition 3

A "series-parallel network function" is a network function at least one of whose n*-switch realizations is a series-parallel network.*

The network of Fig. 1 is series-parallel, hence the corresponding function is a series-parallel network (SPN) function.

In what follows we will be concerned with the characterization of SPN functions and the synthesis of the networks to which they correspond. It should be noted that, although series-parallel networks comprise a small subclass of the class of all relay contact networks,[3] there are other types of switching elements for which they represent the only electrically useful networks.

II. GENERAL THEORY

Theorem 1

The inverse of any SPN function is also an SPN function.

Proof: A Boolean expression may be derived which corresponds explicitly

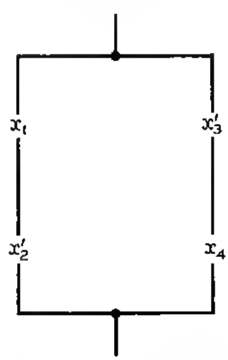

Fig. 1 — Four-switch realization.

to the series-parallel realization of the function.[4] DeMorgan's law, when applied to this expression, yields a Boolean expression which exactly corresponds to a series-parallel n-switch realization of the inverse function.

Theorem 1 may be extended to cover all network functions whose n-switch realizations are planar.[5]

Theorem 2

A minimum generating set of network functions of* n *variables may be put into a one-to-one correspondence with the distinct† networks of* n *elements.* A proof of this theorem has been given by Ashenhurst.[6] Some implications should be noted explicitly: (a) the n-switch realization of a network function is unique, and (b) only network functions correspond to networks.

Theorem 3

Suppose that $\mathrm{f}(\mathrm{x}_1, \mathrm{x}_2, \cdots, \mathrm{x}_n)$ *is an SPN function. Then* f *can be expressed as an SPN function of the variables* $\alpha_1, \alpha_2, \cdots, \alpha_k$, *where each* α *is itself an SPN function of the variables in one of* k *nonoverlapping subsets of the variables* $\mathrm{x}_1, \mathrm{x}_2, \cdots, \mathrm{x}_n$.
Proof: This is obvious from the definitions of SPN function and series-parallel network.

Theorem 4

Let $\mathrm{f}(\mathrm{x}_1, \mathrm{x}_2, \cdots, \mathrm{x}_n)$ *be an SPN function which in canonical form has* m *terms. Let* f *be expressed as an SPN function* $\mathrm{F}(\alpha_1, \alpha_2, \cdots, \alpha_k)$, $\mathrm{k} \leqq \mathrm{n}$, *where each* α_i *is itself an SPN function of* S_i *variables with* m_i *terms in its canonical form (as a function of its* S_i *variables only). In the canonical form of* $\mathrm{F}(\alpha_i)$, *replace each* α_i *by* m_i, *each* α_i' *by* $\mathrm{m}_i' = 2^{\mathrm{S}_i} - \mathrm{m}_i$, *each Boolean* $\oplus$ *by algebraic* $+$ *and each Boolean* $\cdot$ *by algebraic* $\cdot$; *then the result is* m.
Proof: It is only necessary to prove the theorem for two variables, since Definition 1 and Theorem 3 may then be used to complete an inductive proof. Suppose f equals $\alpha_1\alpha_2$.‡ Each term of the canonical form of f may be formed by selecting one of the m_1 terms of α_1 and one of the m_2 terms

* The group of variable transformations (permutations and inversions of the variables) divides any class of switching functions into N nonoverlapping subsets, called equivalence classes; a set of N functions, one from each equivalence class, is called a "minimum generating set."

† Two networks are "distinct" if one is not derivable from the other by transformations consisting of the reversal of two-terminal subnetworks.

‡ Such a function is said to be "essentially series."

of α_2. All such terms must appear, and no others can. Hence m equals m_1m_2.

Now suppose f equals $\alpha_1 \oplus \alpha_2$.* For each of the m_1 terms of α_1 there are 2^{S_2} configurations of the variables of α_2 which appear in the canonical form of f; similarly, 2^{S_1} configurations of the variables of α_1 are associated with each of the m_2 terms of α_2. This gives $2^{S_1}m_2 + 2^{S_2}m_1$ terms. Of these, m_1m_2 are duplicates. Hence

$$m = 2^{S_1}m_2 + 2^{S_2}m_1 - m_1m_2,$$
$$= m_1m_2' + m_1'm_2 + m_1m_2.$$

Theorem 5

If f(x$_1$, $\cdots$, x$_n$) *is any SPN function, and* m *is the number of terms in its canonical form, then* m *is odd.*

Proof: By induction on n. The theorem may be verified by inspection for the cases $n = 1$ and $n = 2$.

Now assume the theorem true for all $k < n$. Consider an arbitrary SPN function of n variables. It can be expressed either as $\alpha_1\alpha_2$ or as $\alpha_1 \oplus \alpha_2$, where α_1 and α_2 are SPN functions of fewer than n variables. Again let m_1 be the number of terms of α_1 and m_2 the number of terms of α_2. Then

$$m = m_1 \cdot m_2$$

is odd, or

$$m = 2^{S_2}m_1 + 2^{S_1}m_2 - m_1m_2$$

is odd, since m_1 and m_2 are odd by the inductive hypothesis. This proves the theorem.

Corollary

It follows immediately that any network function for which m *is even has only a bridge circuit realization.*

As an example, consider the function $f(x_1, x_2, x_3, x_4, x_5)$, whose canonical-form matrix is given in Fig. 2 ($m = 16$). It has the five-switch realization shown, *but no series-parallel five-switch realization.*†

It is reasonable to ask whether an odd number of terms in the canonical form of a network function implies that its realization is series-parallel. This is not so, as may be seen from the following example.

* Such a function is said to be "essentially parallel."

† Note that Theorem 2 asserts that there is *no* other five-switch realization of this function.

The function $F(x_1,x_2,x_3,x_4,x_5,x_6,x_7)$, whose canonical-form matrix and seven-switch realization are shown in Fig. 3, has $m = 59$ but the network is not series-parallel.

Theorem 6[7, 8]

The function $f(x_1, x_2, \cdots, x_n)$ *defined by* $f_i = 1$,* $i = 0, 1, \cdots, m - 1$ *and* $f_i = 0$ *for* $i \geqq m$ *is an SPN function for* m *odd.*

x_1	x_2	x_3	x_4	x_5
0	0	0	0	0
0	0	0	0	1
0	0	0	1	0
0	0	0	1	1
0	0	1	0	0
0	0	1	0	1
0	0	1	1	0
0	0	1	1	1
0	1	0	0	0
0	1	0	0	1
0	1	0	1	0
1	0	0	0	0
1	0	0	0	1
1	0	1	0	0
1	1	0	0	0
1	1	0	0	1

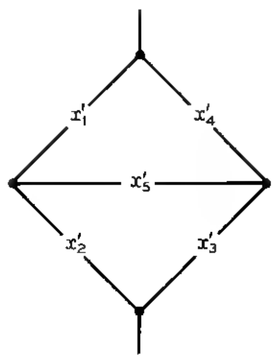

Fig. 2 — Five-switch realization.

Definition 4

The SPN function defined by $f_i = 1$, $i = 0, 1, \cdots, m - 1$ *and* $f_i = 0$ *for* $i \geqq m$, m *odd, is called a "canonical-form network function."*
All SPN functions of one, two and three variables are, to within a variable transformation, canonical-form network functions. The first SPN functions which are not canonical-form network functions appear in four variables. Such functions may be characterized by the following discussion.

Consider the Boolean expression for an SPN function which explicitly represents[4] the n-switch series-parallel network for the function. Then

i. There may exist a variable x such that $f = xg$, with g an SPN function.

* Here, f_i is the value of the function for the configuration of the variables which would represent the integer i in the dyadic number system.

ii. There may exist a variable x such that $f = x \oplus g$, with g an SPN function.

iii. The function may be such that no x exists which satisfies either i or ii.

In cases i and ii the function is said to be "reducible"; in case iii, "irreducible."

If a function is reducible, then the "residue" function g may be examined for reducibility. For any SPN function such reduction will ultimately lead to a residue function which is (a) a single variable, or (b) irreducible and a function of more than one variable. An SPN function whose ultimate residue function is a single variable is said to be "completely reducible."

Theorem 7

An SPN function is, to within a variable transformation, a canonical-form network function if and only if it is completely reducible.

Proof: By induction on n, the number of variables. For $n = 1$ and $n = 2$ the theorem is obviously true.

Now assume the theorem true for all $k < n$. Suppose f to be an SPN

x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_1	x_2	x_3	x_4	x_5	x_6	x_7
0	0	0	0	0	0	0	0	0	1	1	1	0	0	1	0	0	0	0	1	0
0	0	0	0	0	0	1	0	0	1	1	1	1	0	1	0	0	0	1	0	0
0	0	0	0	0	1	0	0	1	0	0	0	0	0	1	0	0	0	1	0	1
0	0	0	0	1	0	0	0	1	0	0	0	0	1	1	0	0	0	1	1	0
0	0	0	0	1	0	1	0	1	0	0	0	1	0	1	0	0	1	0	0	0
0	0	0	0	1	1	0	0	1	0	0	1	0	0	1	0	0	1	0	0	1
0	0	0	1	0	0	0	0	1	0	0	1	0	1	1	0	0	1	0	1	0
0	0	0	1	0	0	1	0	1	0	0	1	1	0	1	0	0	1	1	0	0
0	0	0	1	0	1	0	0	1	0	1	0	0	0	1	0	0	1	1	1	0
0	0	0	1	1	0	0	0	1	0	1	0	0	1	1	0	1	0	0	0	0
0	0	0	1	1	1	0	0	1	0	1	0	1	0	1	0	1	0	0	0	1
0	0	1	0	0	0	0	0	1	0	1	1	0	0	1	0	1	0	0	1	0
0	0	1	0	0	0	1	0	1	0	1	1	1	0	1	0	1	0	1	0	0
0	0	1	0	0	1	0	0	1	1	0	0	0	0	1	0	1	0	1	1	0
0	0	1	0	1	0	0	0	1	1	0	0	0	1	1	0	1	1	0	0	0
0	0	1	0	1	0	1	0	1	1	0	0	1	0	1	0	1	1	0	0	1
0	0	1	0	1	1	0	0	1	1	0	1	0	0	1	0	1	1	0	1	0
0	0	1	1	0	0	0	0	1	1	0	1	0	1	1	0	1	1	1	0	0
0	0	1	1	0	0	1	1	0	0	0	0	0	0	1	0	1	1	1	1	0
0	0	1	1	0	1	0	1	0	0	0	0	0	1							

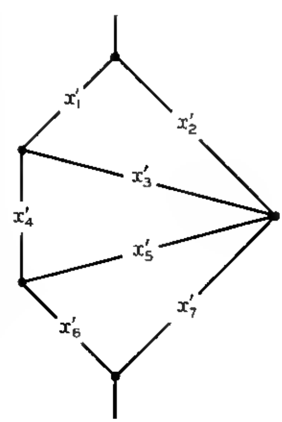

Fig. 3 — Seven-switch realization.

function of n variables which is completely reducible. Then either $f = x'g$ or $f = x' \oplus g$, and g is a completely reducible SPN function of $n - 1$ variables. The inductive hypothesis then guarantees that f is a canonical-form network function.

Suppose now that f is a canonical-form network function of m terms.

(a) If $m < 2^{n-1}$, then f equals $x_n'g(x_1, \cdots, x_{n-1})$, where g is a canonical-form network function.

(b) If $m > 2^{n-1}$, then f equals $x_n' \oplus g(x_1, \cdots, x_{n-1})$, where g is a canonical-form network function.

In both cases, the inductive hypothesis guarantees that g is completely reducible, hence so is f. This proves the theorem.

The irreducible SPN functions of n variables are a set of some interest. One question occurs immediately: How many such functions are there? This may be answered as follows.

Let S_n be the number of functions in a minimum generating set of SPN functions of n variables. Since, by Theorem 2, the correspondence between functions and networks is one-to-one, S_n is the number of series-parallel networks of n elements.* Let i_n be the number of functions in a minimum generating set of irreducible SPN functions of n variables.

Theorem 8

$$i_{n+1} = S_{n+1} - 2S_n .$$

Proof: S_n is the number of series-parallel networks of n elements, and i_n is the number of series-parallel networks of n elements which have no single elements in series or in parallel with the entire network. Form networks of $n + 1$ elements by adding a single element in series to each of the S_n networks of n elements; form the corresponding set by adding one element in parallel. There are $2S_n$ such $n + 1$ element networks. None of these corresponds to an irreducible function. If the networks corresponding to all the irreducible functions are now adjoined, giving $2S_n + i_{n+1}$, this includes all the SPN functions of $n + 1$ variables, since any reducible function is included in the set of $2S_n$ functions, and all irreducible functions are included in the set of i_{n+1} functions. Hence $S_{n+1} = 2S_n + i_{n+1}$. It is obvious that there are 2^{n-1} completely reducible SPN functions of n variables since they are canonical-form network functions, and these correspond to the odd integers m, $1 \leqq m \leqq 2^n$. The remaining SPN functions of n variables are each either irreducible or reducible to some irreducible function. Theorem 9 exhibits this division.

* These numbers are discussed in Ref. 3.

Theorem 9

$$S_n = 2^{n-1} + i_n + \sum_{j=1}^{n-2} i_{j+1}2^{n-j-1}.$$

(*a*) The term 2^{n-1} represents the canonical-form network functions of n variables.
(*b*) The term i_n represents the irreducible SPN functions of n variables.
(*c*) The remaining terms

$$\sum_{j=1}^{n-2} i_{j+1}2^{n-j-1}$$

represent the SPN functions which are reducible to irreducible functions of more than one variable. To prove Theorem 9, it is only necessary to solve[9] the difference equation, $S_{n+1} - 2S_n = i_{n+1}$. Table I shows the numbers of interest for $n = 1(1)10$.

Let $f(x_i)$ be a function of n variables, represented by a canonical-form matrix of zeros and ones (n columns, m rows), as in Section I. Let m_{i0} and $m_{i1} = m - m_{i0}$ be equal to the number of zeros and ones respectively in the x_i column of the matrix.

Theorem 10

If $f(x_i)$ *is an SPN function, it is a function of* x_j' *or* x_j *according as* m_{j1} *or* m_{j0} *is the lesser.*

TABLE I

n	S_n	i_n	$r_n = \sum_{j=1}^{n-2} i_{j+1}2^{n-j-1}$	C_n
1	1	0	0	1
2	2	0	0	2
3	4	0	0	4
4	10	2	0	8
5	24	4	4	16
6	66	18	16	32
7	180	48	68	64
8	522	162	232	128
9	1532	488	788	256
10	4624	1560	2552	512

S_n = total number (taken from Ref. 3) of SPN functions in a minimum generating set.
i_n = number of irreducible SPN functions in a minimum generating set.
r_n = number of reducible but not completely reducible SPN functions in a minimum generating set.
C_n = number of completely reducible SPN functions in a minimum generating set.

Proof: By hypothesis, f can be written as $f = x_j^*A \oplus B$, where (a) x_j^* is either x_j or x_j'; (b) A and B are functions of the remaining variables only. Then

$$f = x_j^*A \oplus x_j^*B \oplus x_j^{*\prime}B$$
$$= x_j^*(A \oplus B) \oplus x_j^{*\prime}B.$$

Since $B \leqq A \oplus B$, the number of terms in B is equal to or less than the number of terms in $A \oplus B$. Now suppose that x_j^* be equal to x_j, then $f = x_j(A \oplus B) \oplus x_j'B$. But the number of terms in B is m_{j0}, and the number of terms in $(A \oplus B)$ expanded as a function of all $n - 1$ variables is m_{j1}, hence $m_{j0} \leqq m_{j1}$. But $m = m_{j0} + m_{j1}$ is odd, hence $m_{j0} < m_{j1}$.

Similarly, if $x_j^* = x_j'$, m_{j0} is the number of terms in $(A \oplus B)$ and m_{j1} is the number of terms in B, so that $m_{j1} \leqq m_{j0}$. Again, the equality is impossible, so that $m_{j1} < m_{j0}$. This completes the proof.

As an example, consider the series-parallel network shown in Fig. 4. The corresponding function is

$$f(x) = x_1'(x_4 \oplus x_2'x_3),$$

and the canonical-form matrix for f is:

x_1	x_2	x_3	x_4
0	0	0	1
0	0	1	0
0	0	1	1
0	1	0	1
0	1	1	1

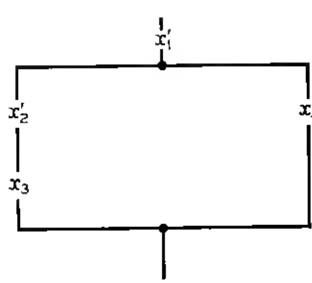

Fig. 4 — Network for $f(x) = x_1'(x_4 \oplus x_2'x_3)$.

The counts for each element are:

j	m_{j0}	m_{j1}
1	5	0
2	3	2
3	2	3
4	1	4

By Theorem 10, f is a function of x_1', x_2', x_3 and x_4, and this is clearly true.

Consider any element x_j in a series-parallel network. Let A_1 represent all of that part of the network which is in parallel with x_j. Let B_1 represent all of that part of the network which is in series with the parallel combination of A_1 and x_j. Let g_1 be equal to $B_1(A_1 \oplus x_j)$. In general, let A_i be that part of the network which is in parallel with g_{i-1}, and B_i be that part of the network which is in series with the parallel combination of A_i and g_{i-1}. Then an SPN function f may be expressed as:

$$f = x_j \prod_{i=1}^{N} B_i \oplus \sum_{k=1}^{N} A_k \prod_{i=k}^{N} B_i,^*$$

where A_i and B_i are SPN functions of distinct variables (see Fig. 5); A_1 may be the trivial function 0, while B_N may be the trivial function 1.

Definition 5

x_j *is said to be a "series" element if and only if* $A_1 \equiv 0$, *and* x_j *is said to be a "parallel" element if and only if* $A_1 \not\equiv 0$.

Theorem 11

In the circuit for an SPN function, x_j *is a series (parallel) element if and only if the lesser of* m_{j0} *and* m_{j1} *is even (odd).*

Proof: Suppose that f is expressed as:

$$f = x_j \prod_{i=1}^{N} B_i \oplus \sum_{k=1}^{N} A_k \prod_{i=k}^{N} B_i,$$

with A_i and B_i SPN functions of nonoverlapping subsets of the vari-

* Henceforth the symbol Σ will be used to signify extended application of the operation $\oplus$.

ables. Then

$$f(x_j = 1, A_i, B_i) = \prod_{i=1}^{N} B_i \oplus \sum_{k=1}^{N} A_k \prod_{i=k}^{N} B_i$$

$$= \prod_{i=1}^{N} B_i \oplus A_1 \prod_{i=1}^{N} B_i \oplus \sum_{k=2}^{N} A_k \prod_{i=k}^{N} B_i .$$

Clearly A_1 is a vacuous variable in this function, and wherever A_1P appears in the canonical form $A_1'P$ appears also (P is a product involving $A_2, \cdots, A_N, B_1, \cdots, B_N$ and their primes). Hence, if m_{A_1} is the number of terms in A_1 and A_1 is a function of S_{A_1} variables, then, by Theorem 4, $m_{A_1} + m_{A_1}' = 2^{S_{A_1}}$ is a factor of m_{j1}. If $A_1 \neq 0$, then $S_{A_1} \neq 0$, hence m_{j1} is even and m_{j0} odd.

Now suppose A_1 is identically equal to 0, then

$$f(x_j = 0, A_i, B_i) = \sum_{k=1}^{N} A_k \prod_{i=k}^{N} B_i$$

$$= A_1 \prod_{i=1}^{N} B_i \oplus \sum_{k=2}^{N} A_k \prod_{i=k}^{N} B_i$$

$$= \sum_{k=2}^{N} A_k \prod_{i=k}^{N} B_i ,$$

and clearly B_1 is vacuous. Hence, $m_{B_1} + m_{B_1}' = 2^{S_{B_1}}$ is a factor of m_{j0},

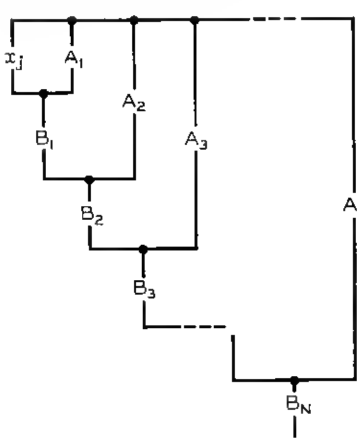

Fig. 5 — Network for $f = x_j \prod_{i=1}^{N} B_i \oplus \sum_{k=1}^{N} A_k \prod_{i=k}^{N} B_i$

and m_{j0} is even, m_{j1} odd. (Note that $S_{A_1} = S_{B_1} = 0$ implies $f \equiv 0$; this case need not be considered.)

Theorem 12

If two elements x_j *and* x_k *are in series or in parallel, then* m_{j0} *equals* m_{k0} *and* m_{j1} *equals* m_{k1} .

Proof: If the function is expanded as

$$f = x_j x_k \prod_{i=1}^{N} B_i \oplus \sum_{k=1}^{N} A_k \prod_{i=k}^{N} B_i$$

or as

$$f = (x_j \oplus x_k) \prod_{i=1}^{N} B_i \oplus \sum_{k=1}^{N} A_k \prod_{i=k}^{N} B_i ,$$

and Theorem 4 is applied, the result follows immediately.

It should be noted that the converse is not true; that is, $m_{j0} = m_{k0}$ and $m_{j1} = m_{k1}$ does not necessarily imply that x_j and x_k are in series or in parallel. This will be seen in one of the examples which will be given.

Theorem 13

One of m_{j0} *and* m_{j1} *is even; call this* $\bar{m}$. *Let* k *be the largest integer such that* $2^k \mid \bar{m}$.* *Then* k *is the number of switches in the subnetwork which is in series or in parallel with* x_j .

Proof: Suppose that f is a function of x_j (not of x_j'). This assumption is irrelevant in the argument, and is made only for the sake of definiteness. Then $m_{j0} < m_{j1}$.

Case 1: Suppose that x_j is a parallel element. Then m_{j0} is odd, by Theorem 11, hence $\bar{m} = m_{j1}$. Clearly, in the function $f(x_j = 1, A_i , B_i)$, A_1 is a vacuous variable. Therefore, in the canonical form of $f(x_j = 1, A_i , B_i)$ wherever $A_1'P$ appears so does A_1P. Hence, by Theorem 4,

$$\begin{aligned} m_{j1} &= (m_{A_1} + m_{A_1}')C \\ &= 2^{S_{A_1}}C. \end{aligned}$$

But C is odd (if it were even, then removing the subfunction A_1 would make x_j a series element, with m_{j0} odd, and this is impossible by Theorem 11). Therefore $2^{S_{A_1}} \mid \bar{m}$, and $2^{(S_{A_1}+k)} \nmid \bar{m}$ for $k = 1, 2, \cdots$.

Case 2: If x_j is supposed a series element, then clearly $\bar{m} = m_{j0}$. Also B_1 is vacuous in $f(x_j = 0, A_i , B_i)$. (Note that $A_1 \equiv 0$ by hypoth-

*a | b as usual means "a divides b"; a ∤ b means "a does not divide b."

esis.) A similar argument shows that $2^{S_{B_1}} \mid \bar{m}$, while $2^{(S_{B_1}+k)} \nmid \bar{m}$ for $k = 1, 2, \cdots$. This proves the theorem.

Theorem 14

Consider an SPN function f. *Of* f *and* f′, *one is essentially series, say* f. *Then* f *may be expressed as* f $= \alpha \cdot \beta$, *where* α, β *are* SPN *functions of nonoverlapping subsets of the variables. If* α *equals* $\alpha(x_j)$, *then* $m_\beta \mid m_{j0}$ *and* $m_\beta \mid m_{j1}$; *if* β *equals* $\beta(x_j)$, *then* $m_\alpha \mid m_{j0}$ *and* $m_\alpha \mid m_{j1}$.
Proof: If α equals $\alpha(x_j)$ then $m_\beta \mid m_{j0}$ and $m_\beta \mid m_{j1}$, since every term in the canonical form of α (as a function of its subset of the variables) is repeated in the canonical form of f exactly m_β times.

III. SYNTHESIS

Consider the function defined by tbe canonical-form matrix:

x_1	x_2	x_3	x_4	x_5
0	1	0	1	1
0	1	1	0	0
0	1	1	0	1
0	1	1	1	0
0	1	1	1	1
1	0	0	1	1
1	0	1	0	0
1	0	1	0	1
1	0	1	1	0
1	0	1	1	1
1	1	0	1	1
1	1	1	0	0
1	1	1	0	1
1	1	1	1	0
1	1	1	1	1

Suppose it is given that this function is an SPN function, and it is required to find the network.

Table II shows the counts for each variable and the information which is derived from these counts by application of Theorems 10 through 14.

Table II

j	m_{j0}	m_{j1}	Variable	Series or Parallel	k	Prime Factorization of	
						m_{j0}	m_{j1}
1	5	10	x_1	P	1	5	2, 5
2	5	10	x_2	P	1	5	2, 5
3	3	12	x_3	P	2	3	3, 2^2
4	6	9	x_4	S	1	2, 3	3^2
5	6	9	x_5	S	1	2, 3	3^2

The column headed "variable" is filled in by applying Theorem 10. Since in every case $m_{j0} < m_{j1}$, only unprimed variables occur. The column headed "series or parallel" is filled in by applying Theorem 11. Since m_{10}, m_{20} and m_{30} are odd, x_1, x_2 and x_3 are parallel elements; x_4 and x_5 are series elements, since m_{40} and m_{50} are even. The column headed "k" is filled in for use in applying Theorem 13. Since $2 \mid 10$ and $4 \nmid 10$, the values for x_1 and x_2 are 1. Also $4 \mid 12$ and $8 \nmid 12$ gives $k = 2$ for x_3. Similarly for x_4 and x_5 the values are 1. The last two columns give the prime factors of m_{j0} and m_{j1}. Since $m_{11} = m_{21} = 10$ and each of x_1 and x_2 is in parallel with a single switch and since no other m_{j1} equals 10, x_1 and x_2 are in parallel, i.e., $(x_1 \oplus x_2)$ represents a component of the network. Similarly $(x_4 \cdot x_5)$ represents a component. Since x_3 is in parallel with a two-switch network but not in parallel with x_1 (by Theorem 12), it must be in parallel with $x_4 \cdot x_5$. Hence $(x_3 \oplus x_4x_5)$ represents a component. Since x_3 cannot be in parallel with x_1, these two components must be in series, therefore $f = (x_1 \oplus x_2)(x_3 \oplus x_4x_5)$ and the network is as shown in Fig. 6. Note that if α equals $(x_1 \oplus x_2)$ and β equals $(x_3 \oplus x_4x_5)$, then $m_\alpha = 3$, $m_\beta = 5$ and

$$m_\beta \mid m_{10}, \quad m_\beta \mid m_{11}, \quad m_\beta \mid m_{20}, \quad m_\beta \mid m_{21}, \quad m_\beta \mid m_{30},$$
$$m_\alpha \mid m_{31}, \quad m_\alpha \mid m_{40}, \quad m_\alpha \mid m_{41}, \quad m_\alpha \mid m_{50}, \quad m_\alpha \mid m_{51},$$

as required by Theorem 14. If Theorems 10 through 14 are to be applied mechanically, the following fact should be noted. Once it has been deter-

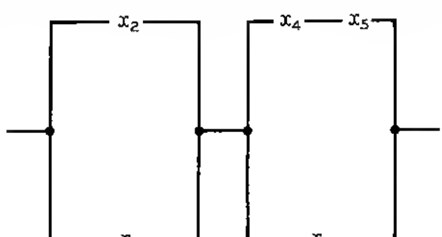

Fig. 6 — Network for $f = (x_1 \oplus x_2)(x_3 \oplus x_4x_5)$.

mined that $(x_1 \oplus x_2)$ and x_4x_5 represent components, these combinations may be treated as single elements; the canonical form may then be modified and the count repeated. For example, let A equal $(x_1 \oplus x_2)$ and B equal x_4x_5. Then the original canonical form may be rewritten (in two steps) as

A	x_3	x_4	x_5
1	0	1	1
1	1	0	0
1	1	0	1
1	1	1	0
1	1	1	1

A	x_3	B
1	0	1
1	1	0
1	1	1

and this function is obviously $A(x_3 \oplus B)$.

Consider the canonical-form matrix:

x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9
0	0	1	1	0	0	0	0	0
0	0	1	1	0	0	0	0	1
0	0	1	1	0	0	0	1	1
0	0	1	1	0	0	1	0	0
0	0	1	1	0	0	1	0	1
0	0	1	1	0	0	1	1	1
0	0	1	1	0	1	1	0	0
0	0	1	1	0	1	1	0	1
0	0	1	1	0	1	1	1	1
0	0	1	1	1	0	1	0	0
0	0	1	1	1	0	1	0	1
0	0	1	1	1	0	1	1	1
0	0	1	1	1	1	1	0	0
0	0	1	1	1	1	1	0	1
0	0	1	1	1	1	1	1	1
0	1	1	1	0	0	0	0	0
0	1	1	1	0	0	0	0	1
0	1	1	1	0	0	0	1	1
0	1	1	1	0	0	1	0	0
0	1	1	1	0	0	1	0	1
0	1	1	1	0	0	1	1	1
0	1	1	1	0	1	1	0	0
0	1	1	1	0	1	1	0	1
0	1	1	1	0	1	1	1	1
0	1	1	1	1	0	1	0	0
0	1	1	1	1	0	1	0	1
0	1	1	1	1	0	1	1	1
0	1	1	1	1	1	1	0	0
0	1	1	1	1	1	1	0	1
0	1	1	1	1	1	1	1	1
1	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	1
1	0	0	0	0	0	0	1	1
1	0	0	0	0	0	1	0	0
1	0	0	0	0	0	1	0	1

x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9
1	0	0	0	0	0	1	1	1
1	0	0	0	0	1	1	0	0
1	0	0	0	0	1	1	0	1
1	0	0	0	0	1	1	1	1
1	0	0	0	1	0	1	0	0
1	0	0	0	1	0	1	0	1
1	0	0	0	1	0	1	1	1
1	0	0	0	1	1	1	0	0
1	0	0	0	1	1	1	0	1
1	0	0	0	1	1	1	1	1
1	0	0	1	0	0	0	0	0
1	0	0	1	0	0	0	0	1
1	0	0	1	0	0	0	1	1
1	0	0	1	0	0	1	0	0
1	0	0	1	0	0	1	0	1
1	0	0	1	0	0	1	1	1
1	0	0	1	0	1	1	0	0
1	0	0	1	0	1	1	0	1
1	0	0	1	0	1	1	1	1
1	0	0	1	1	0	1	0	0
1	0	0	1	1	0	1	0	1
1	0	0	1	1	0	1	1	1
1	0	0	1	1	1	1	0	0
1	0	0	1	1	1	1	0	1
1	0	0	1	1	1	1	1	1
1	0	1	0	0	0	0	0	0
1	0	1	0	0	0	0	0	1
1	0	1	0	0	0	0	1	1
1	0	1	0	0	0	1	0	0
1	0	1	0	0	0	1	0	1
1	0	1	0	0	0	1	1	1
1	0	1	0	0	1	1	0	0
1	0	1	0	0	1	1	0	1
1	0	1	0	0	1	1	1	1
1	0	1	0	1	0	1	0	0

x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9
1	0	1	0	1	0	1	0	1
1	0	1	0	1	0	1	1	1
1	0	1	0	1	1	1	0	0
1	0	1	0	1	1	1	0	1
1	0	1	0	1	1	1	1	1
1	0	1	1	0	0	0	0	0
1	0	1	1	0	0	0	0	1
1	0	1	1	0	0	0	1	1
1	0	1	1	0	0	1	0	0
1	0	1	1	0	0	1	0	1
1	0	1	1	0	0	1	1	1
1	0	1	1	0	1	1	0	0
1	0	1	1	0	1	1	0	1
1	0	1	1	0	1	1	1	1
1	0	1	1	1	0	1	0	0
1	0	1	1	1	0	1	0	1
1	0	1	1	1	0	1	1	1
1	0	1	1	1	1	1	0	0
1	0	1	1	1	1	1	0	1
1	0	1	1	1	1	1	1	1
1	1	1	1	0	0	0	0	0
1	1	1	1	0	0	0	0	1
1	1	1	1	0	0	0	1	1
1	1	1	1	0	0	1	0	0
1	1	1	1	0	0	1	0	1
1	1	1	1	0	0	1	1	1
1	1	1	1	0	1	1	0	0
1	1	1	1	0	1	1	0	1
1	1	1	1	0	1	1	1	1
1	1	1	1	1	0	1	0	0
1	1	1	1	1	0	1	0	1
1	1	1	1	1	0	1	1	1
1	1	1	1	1	1	1	0	0
1	1	1	1	1	1	1	0	1
1	1	1	1	1	1	1	1	1

TABLE III

j	m_{j0}	m_{j1}	Variable	S or P	k	Prime Factorization of	
						m_{j0}	m_{j1}
1	30	75	x_1	S	1	2, 3, 5	$3, 5^2$
2	75	30	x_2'	S	1	$3, 5^2$	2, 3, 5
3	30	75	x_3	S	1	2, 3, 5	$3, 5^2$
4	30	75	x_4	S	1	2, 3, 5	$3, 5^2$
5	63	42	x_5'	S	1	$3^2, 7$	2, 3, 7
6	63	42	x_6'	S	1	$3^2, 7$	2, 3, 7
7	21	84	x_7	P	2	3, 7	$2^2, 3, 7$
8	70	35	x_8'	P	1	2, 5, 7	5, 7
9	35	70	x_9	P	1	5, 7	2, 5, 7

This gives Table III. It is evident that

i. one of x_1x_2', x_1x_3, x_1x_4 appears in the network;

ii. $x_5'x_6'$ appears in the network;

iii. $x_8' \oplus x_9$ appears in the network.

In order to select the correct one of x_1x_2', x_1x_3, x_1x_4, Table IV is constructed. Note that if xy is to be a component it must be possible to substitute A for xy in the canonical form. Then a count can be made of m_{A0} and m_{A1}. But A equals zero implies three terms as shown:

$A = 1$

xy
11

$A = 0$

xy
00
01
10

Therefore, the number of occurrences of $xy = 00$, $xy = 01$ and $xy = 10$ must all be equal. Clearly, in the example only the combination x_1x_2' satisfied this requirement. This implies that x_1x_2' and x_3x_4 represent

TABLE IV

	00	01	10	11
x_1x_2	15	15	60	15
x_1x_3	0	30	30	45
x_1x_4	0	30	30	45

components. Reducing the canonical form in terms of the new variables

$$y_1 = x_1x_2'$$

$$y_2 = x_3x_4$$

$$y_3 = x_5'x_6'$$

$$y_4 = x_7$$

$$y_5 = x_8' \oplus x_9$$

gives the following matrix:

y_1	y_2	y_3	y_4	y_5
0	1	0	1	1
0	1	1	0	1
0	1	1	1	1
1	0	0	1	1
1	0	1	0	1
1	0	1	1	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1

Analysis of the counts for this matrix is shown in Table V. A count for y_1y_2, y_1y_3, y_1y_4 yields Table VI. Therefore $z_1 = x_1x_2' \oplus x_3x_4$, $z_2 = x_5'x_6' \oplus x_7$, and $z_3 = x_8' \oplus x_9$ represent components. Again reducing the canonical-form matrix leads to

z_1	z_2	z_3
1	1	1

Hence the function is

$$f = (x_1x_2' \oplus x_3x_4)\,(x_5'x_6' \oplus x_7)\,(x_8' \oplus x_9)$$

and the network is as shown in Fig. 7.

It should be noted that an attempt to apply Theorems 10 through 14 to a non-SPN function will lead to a contradiction. For example, the function whose canonical-form matrix and bridge network are shown in

TABLE V

j	m_{j0}	m_{j1}	Variable	S or P	k	Prime Factorization of	
						m_{j0}	m_{j1}
1	3	6	y_1	P	1	3	3, 2
2	3	6	y_2	P	1	3	3, 2
3	3	6	y_3	P	1	3	3, 2
4	3	6	y_4	P	1	3	3, 2
5	0	9	y_5	S	—	—	3^2

TABLE VI

	00	01	10	11
y_1y_2	0	3	3	3
y_1y_3	1	2	2	4
y_1y_4	1	2	2	4

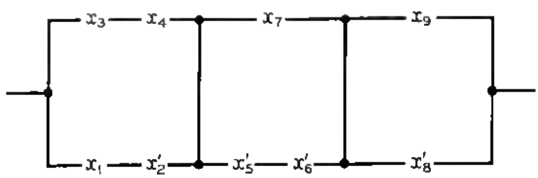

Fig. 7 — Network for $f = (x_1x_2' \oplus x_3x_4)\ (x_5'x_6' \oplus x_7)\ (x_8' \oplus x_9)$.

TABLE VII

j	m_{j0}	m_{j1}	Variable	S or P	k
1	38	21	x_1'	P	1
2	43	16	x_2'	S	4
3	33	26	x_3'	S	1
4	34	25	x_4'	P	1
5	33	26	x_5'	S	1
6	38	21	x_6'	P	1
7	43	16	x_7'	S	4

Fig. 3 yields Table VII. If this function is to be an SPN function, x_4' must be in parallel with a one-element subnetwork. But no other element exists with the number pair (34,25) as (m_{j0}, m_{j1}), and this contradicts Theorem 12.

The results produced thus far by this investigation suggest many intriguing possibilities. For example, if a function is not an SPN function because it requires both x_j and x_j' contacts, can rules be derived for

augmenting the canonical form of the given function so as to produce the canonical form of a new function which (a) is an SPN function of $n + 1$ variables and (b) reduces to the given function when the new variable is replaced by x_j'? Obviously this notion can be extended to cover all functions by introducing a large enough number of new variables. That this is not a trivial question is shown by the following example:

Let $f(x_1, x_2, x_3, x_4)$ be defined by the canonical-form matrix

x_1	x_2	x_3	x_4
0	0	0	0
0	0	0	1
0	0	1	0
0	0	1	1
0	1	0	0
1	1	0	0

(1)

Since $m = 6$, it is clear that this function cannot be an SPN function. Let us assume however, that it can be realized using five switches including both an x_2 and an x_2', and attempt to form a function $g(x_1, x_2, x_3, x_4, a)$ such that (i) g is an SPN function, and (ii) $g(x_1, x_2, x_3, x_4, a = x_2')$ equals $f(x_1, x_2, x_3, x_4)$. We proceed as follows:

1. Copy the canonical form matrix of f, adding a column headed a. This column is the inverse of the column headed x_2.

x_1	x_2	x_3	x_4	a
0	0	0	0	1
0	0	0	1	1
0	0	1	0	1
0	0	1	1	1
0	1	0	0	0
1	1	0	0	0

(2)

Condition (ii) requires that these terms appear in the function g.

2. The canonical form matrix for f' is similarly augmented, giving

x_1	x_2	x_3	x_4	a
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	1
1	0	0	1	1
1	0	1	0	1
1	0	1	1	1
1	1	0	1	0
1	1	1	0	0
1	1	1	1	0

(3)

and condition (ii) requires that these terms appear in g'.

3. List the remaining possible terms of g. (These will include all combinations for which $a = x_2$.)

x_1	x_2	x_3	x_4	a
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	1
0	1	0	1	1
0	1	1	0	1
0	1	1	1	1
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	0
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1

(4)

Since these terms will vanish when a is set equal to x_2', any of them may appear in the function g. Assuming that g is to be an SPN function of primed variables only, it follows that

(a) whenever, in the function g, the combinations $x_1 0 x_3 x_4 1$ or $x_1 1 x_3 x_4 0$ appear, so must $x_1 0 x_3 x_4 0$, and

(b) the combination $x_1 1 x_3 x_4 1$ may appear in g only if $x_1 0 x_3 x_4 0$, $x_1 0 x_3 x_4 1$ and $x_1 1 x_3 x_4 0$ all appear as well.

4. By the application of these rules the terms in (4) may be divided into three sets: (a) those which must appear in g, (b) those which must appear in g' and (c) those which these rules do not determine. For this

case, these sets are

appear in g

x_1	x_2	x_3	x_4	a
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
1	0	0	0	0

appear in g'

x_1	x_2	x_3	x_4	a
0	1	0	1	1
0	1	1	0	1
0	1	1	1	1
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1

not determined

x_1	x_2	x_3	x_4	a
0	1	0	0	1
1	0	0	1	0
1	0	1	0	0
1	0	1	1	0

5. Combining these with the known terms of g and g' gives

appear in g

x_1	x_2	x_3	x_4	a
0	0	0	0	0
0	0	0	0	1
0	0	0	1	0
0	0	0	1	1
0	0	1	0	0
0	0	1	0	1
0	0	1	1	0
0	0	1	1	1
0	1	0	0	0
1	0	0	0	0
1	1	0	0	0

appear in g'

x_1	x_2	x_3	x_4	a
0	1	0	1	0
0	1	0	1	1
0	1	1	0	0
0	1	1	0	1
0	1	1	1	0
0	1	1	1	1
1	0	0	0	1
1	0	0	1	1
1	0	1	0	1
1	0	1	1	1
1	1	0	0	1
1	1	0	1	0
1	1	0	1	1
1	1	1	0	0
1	1	1	0	1
1	1	1	1	0
1	1	1	1	1

not determined

x_1	x_2	x_3	x_4	a
0	1	0	0	1
1	0	0	1	0
1	0	1	0	0
1	0	1	1	0

The terms presently given as appearing in g form an SPN function

$$g = x_1'x_2' \oplus x_3'x_4'a'$$

and

$$g(x_1, x_2, x_3, x_4, a = x_2') = x_1'x_2' \oplus x_2x_3'x_4' = f(x_1, x_2, x_3, x_4).$$

However, note that if all the terms listed as "not determined" are adjoined to those which "appear in g," a canonical form for the SPN function

$$g(x_1, x_2, x_3, x_4, a) = (x_1' \oplus a')(x_2' \oplus x_3'x_4')$$

is produced, and

$$g(x_1, x_2, x_3, x_4, a = x_2') = (x_1' \oplus x_2)(x_2' \oplus x_3'x_4')$$
$$= f(x_1, x_2, x_3, x_4).$$

From this example, it is clear that the requirement that g be an SPN function, with $g(x_1, x_2, x_3, x_4, a = x_2') = f(x_1, x_2, x_3, x_4)$ does not specify g uniquely.

IV. CONCLUSIONS

Clearly the derivation of the circuit for an SPN function is a process which can be mechanized. If a method for augmenting the canonical form of a non-network function can be formulated, then a purely mechanical process could be set up for deriving the minimum series-parallel networks for an arbitrary function.

REFERENCES

1. Shannon, C. E., A Symbolic Analysis of Relay and Switching Circuits, Trans. A.I.E.E., **57**, December 1938, pp. 713–723.
2. Staff of the Computation Laboratory, Synthesis of Electronic Computing and Control Circuits, Annals Harvard Comp. Lab., **27**, 1951.
3. Riordan, J. and Shannon, C. E., The Number of Two-Terminal Series-Parallel Networks, J. Math. Phys., **21**, August 1942, pp. 83–93.
4. Semon, W. L., The Application of Matrix Methods in the Theory of Switching, Harvard Comp. Lab., Theory of Switching, Report BL-6, May 1954, pp. I-6–I-16.
5. Keister, W., Ritchie, A. E. and Washburn, S. H., *The Design of Switching Circuits*, D. Van Nostrand Co., New York, 1951, pp. 95–99.
6. Ashenhurst, R. L., A Uniqueness Theorem for Abstract Two Terminal Switching Circuits, Harvard Comp. Lab., Theory of Switching, Report BL-10, March 1955, Sec. VI.
7. Quine, W. V., Two Theorems About Truth Functions, Bol. Soc. Math. Mexicana, **10**, 1953, pp. 64–70.
8. Semon, W. L., Adding Transformations, Harvard Comp. Lab., Theory of Switching, Report BL-4, July 1953, Sec. I.
9. Milne-Thomson, L. M., *The Calculus of Finite Differences*, Macmillan, London, 1933.